

Slovem **informatika** označujeme vědní obor, který se zabývá získáváním, zpracováním a využitím informací. Počátky oboru sahají do poloviny 20. století, kdy vznikly první elektronické počítače a začala skutečná informační revoluce. Informatika se zabývá nejen obecnými teoretickými otázkami spojenými s informacemi, ale prostřednictvím řady aplikovaných oborů řeší i praktické problémy.

K nejvýznamnějším oblastem aplikované informatiky patří:

**Umělá inteligence** (AI – Artificial Inteligency). Pokouší se napodobit intelektuální činnosti člověka pomocí počítače (rozpoznávání řeči, tvorba hypotéz).

**Expertní systémy.** Pomáhají odborníkům rozhodovat v oblastech, kde je zapotřebí brát v úvahu velké množství údajů (např. diagnóza nemoci). Do expertního systému se vloží výsledky mnoha různých vyšetření a systém na základě své báze znalostí a složitých algoritmů nabídne možné diagnózy.

**Kybernetika a robotika** se zabývá konstrukcí strojů, které samostatně reagují na podněty. V průmyslové praxi se již celkem běžně využívají průmyslové roboty, experimentálně jsou vyvíjeny humanoidní stroje.

**Počítačová simulace** umožňuje zkoumat neexistující objekty (které je možné nasimulovat, matematicky popsat), případně objekty, které lze jen obtížně zkoumat přímo (vznik hvězdy, průběh tlaků v centru hurikánu atd.).

**Telekomunikace.** Zkoumá a řeší vzdálený přenos informací, výsledkem jsou internet, bezdrátová komunikace, mobilní telefony.

Technické (hardware) i programové (software) prostředky, které jsou pro moderní zpracování informací využívány, bývají souhrnně označovány jako **informační a komunikační technologie** (ICT - Information and Communication Technologies).

**Data** jsou fakta získaná z reality, např. naměřené fyzikální veličiny (teplota, tlak, hmotnost), historické údaje, fotografie, zvukové či filmové záznamy apod.

Pojmem **informace** jsou odborně označována již vyhodnocená a smysluplně uspořádaná data do podoby, která má určitý význam. Díky informacím si vytváříme poznatky o okolním světě a na základě informací se můžeme rozhodovat o svém jednání.

Informace má nehmotnou povahu. Pro její uložení, přenos a zpracování je používán **signál** – fyzický nositel informací.

V primitivní podobě může být signál např. uzel na provázku, vyvěšená vlajka na stěžni apod. Pro strojové zpracování informací se jako signály využívají různé fyzikální veličiny, které mění svůj stav (elektrické napětí, magnetická polarizace, elektromagnetické vlnění, zvuková vlna, tlak plynu nebo tekutiny...).



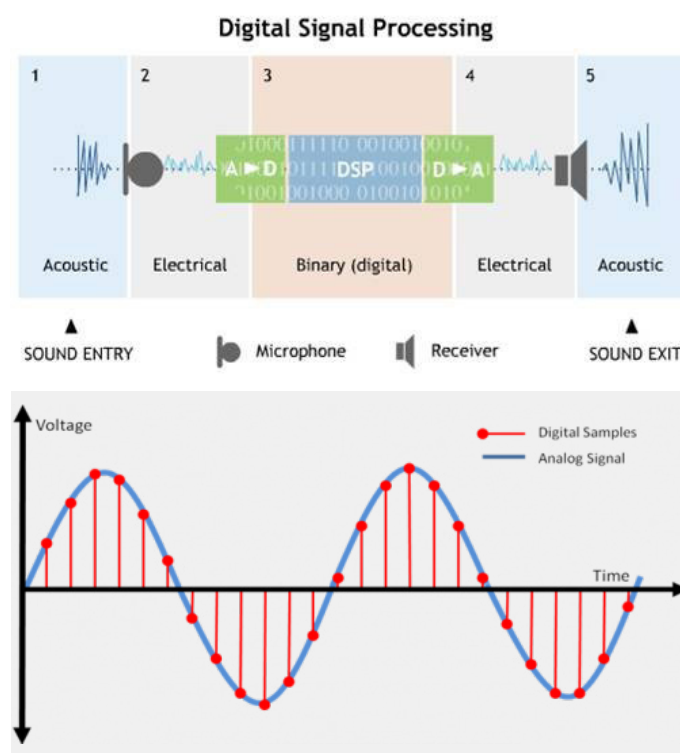
Rozdíl mezi daty a informacemi je možné vysvětlit na příkladu identifikačního průkazu. Průkazka obsahuje textová, číselná, grafická data. Izolovaná data sama o sobě nemají význam, teprve jejich začlenění do průkazky vytváří smysluplnou informaci o daném člověku.

V případě zmíněných fyzikálních jevů lze přirozené změny stavu vyjádřit a zaznamenat v podobě spojitě křivky – analogového signálu.

**Analogový signál** můžeme zpracovat pomocí příslušného analogového zařízení (například analogový magnetofon nahrávající zvuk na magnetickou pásku). Analogový signál věrně napodobuje původní signál, ovšem přenosem a kopírováním původní křivky dochází ke zkreslení (zvýšení šumu) a ztrátě kvality.

K analogovému záznamu zvuku je možné použít mikrofon připojený k analogovému magnetofonu. Zvukové vlny jsou zachyceny mikrofonem, který vstupní audiosignál převede na měnící se elektrické napětí. Elektrický signál je magnetofonem zesílen a snímán nahrávací hlavou. Při průchodu pásky podél hlavy se magnetické částice uspořádají tak, aby odpovídaly směru magnetického pole. Při přehrávání pásky indukuje magnetický záznam elektrický signál ve snímací hlavě; ten se po zesílení přehraje v reproduktoru.

Pro zpracování signálu na digitálním zařízení (např. počítač) je nutné provést jeho **digitalizaci** – vyjádření v číslech (*digitální* = *číselný*), nejčastěji v binární soustavě (sledu jedniček a nul). K digitalizaci se používá **A/D převodník** (analog/digital convertor). Původní křivka analogového signálu je „rozporcována“ na malé úseky (vzorkování) a pro každý z těchto vzorků je zaznamenána číselná hodnota (kvantování). Výsledkem je nespojitý **digitální signál**, který lze mnohonásobně kopírovat bez ztráty kvality.



**Bit** (z anglického *binary digit - dvojková číslice*) je základní a současně nejmenší jednotkou informace. Značí se malým písmenem **b**, např. 16 b. Jeden bit reprezentuje logickou informaci typu ano/ne, kterou můžeme vyjádřit binárními číslicemi 0 a 1. V praxi se s bity setkáme např. při označení architektury systému (32bitový, 64bitový) nebo při určení tzv. **bitové hloubky** - tj. počtu bitů potřebných k uložení jednoho vzorku dat. Jednotkou přenosové rychlosti je **bit za sekundu** - *bit/s* (anglicky *bps – bit per second*); např. linka s přenosovou rychlostí 20 Mbit/s je schopna každou sekundu přenést 20 megabitů dat.

**Byte** (čti bajt, z anglického *byte = slabika*), je jednotka množství dat v informatice. Značí se velkým písmenem **B**, např. 1024 B. Zahrnuje osm bitů, tzn. osmiciferné binární číslo. Do jednoho bajtu je tak možné uložit celkem 256 různých hodnot ( $2^8 = 256$ ). Jeden bajt je obvykle nejmenší objem dat, se kterým dokáže počítač (resp. procesor) přímo pracovat; v bytech a odvozených jednotkách se zpravidla udává kapacita (velikost) paměti.

## ODVOZENÉ JEDNOTKY

Pro bajty i bity se používají tradiční předpony soustavy SI jako kilo, mega, giga atd. (např. 5 GB, 2 Mb/s, 4 KB). Tyto předpony však mají odlišný význam: z technologických důvodů jsou velikosti některých počítačových pamětí obvykle násobkem mocniny dvou - kilobyte nepředstavuje přesně 1 000 bajtů, ale  $2^{10} = 1 024$  bajtů; podobně je to i u dalších odvozených jednotek.

Odvozená jednotka	Značka	Velikost v B	Mocnina
kilobyte	KB	1 024	$2^{10}$
megabyte	MB	1 048 576	$2^{20}$
gigabyte	GB	1 073 741 824	$2^{30}$
terabyte	TB	1 099 511 627 776	$2^{40}$
petabyte	PB	1 125 899 906 842 624	$2^{50}$
exabyte	EB	1 152 921 504 606 846 976	$2^{60}$
zettabyte	ZB	1 180 591 620 717 411 303 424	$2^{70}$

## Číselné soustavy a informatika

Na rozdíl od lidského světa čísel, v němž hraje prim desítková (dekadická) soustava, vládne v digitálním světě **soustava dvojková** (*binární*), která odpovídá i dvěma základním stavům elektronických obvodů:

stav VYPNUTO	stav ZAPNUTO
v obvodu <b>není napětí</b>	v obvodu <b>je napětí</b>
vyjádřeno číslem <b>0</b>	vyjádřeno číslem <b>1</b>
logická <b>NEPRAVDA (false)</b>	logická <b>PRAVDA (true)</b>

V počítačové praxi se používají také soustavy **osmičková** (*oktalová*) a **šestnáctková** (*hexadecimální*). V hexadecimální soustavě jsou chybějící číslice (nad hodnotu 9) vyjadřovány pomocí abecedních znaků (tj. 10 = A, 11 = B, 12 = C, 13 = D, 14 = E, 15 = F). Hexadecimálně se zapisují například adresy v paměti (např. 00H - 1FH), barvy v HTML a CSS (*color="#FFFFFF", color: #FFFFFF*), kódy ve znakových sadách (0x00 až 0x7F) či fyzické MAC adresy síťových zařízení (např. f4:08:a2:36:cc:87).

## Převody mezi číselnými soustavami

**Převeďte desítkové číslo 56 do dvojkové, osmičkové a šestnáctkové soustavy**

Pro převod do dvojkové (binární) soustavy dělíme číslo zapsané v desítkové soustavě základem  $m = 2$  a zapisujeme zbytky (tj. v tomto případě 0 nebo 1):

56 : 2 = 28	28 : 2 = 14	14 : 2 = 7	7 : 2 = 3	3 : 2 = 1	1 : 2 = 0
zbytek 0	zbytek 0	zbytek 0	zbytek 1	zbytek 1	zbytek 1

Zjištěné zbytky uspořádáme v obráceném pořadí - tj. zprava doleva - a dostaneme výslednou binární podobu čísla:  $(56)_{10} = (111000)_2$

Podobně můžeme postupovat i v případě převodu desítkového čísla do osmičkové nebo šestnáctkové soustavy. Změní se pouze základ dělení: v případě osmičkové soustavy dělíme osmi ( $m = 8$ ), v případě šestnáctkové šestnácti ( $m = 16$ ). Zbytky opět zapisujeme zprava:

$(56)_{10} = (70)_8$	56 : 8 = 7	7 : 8 = 0	$(56)_{10} = (38)_{16}$	56 : 16 = 3	3 : 16 = 0
	zbytek 0	zbytek 7		zbytek 8	zbytek 3

**Převeďte binární číslo 11010110 do desítkové soustavy**

Převod z dvojkové do desítkové soustavy můžeme urychlit, když si uvědomíme, že jednotlivé řády (číslíky) binárního čísla představují k-tou mocninu základu - tedy 2. Stačí poté sečíst pouze ty mocniny, které jsou v binárním čísle zastoupeny jedničkou:

1	1	0	1	0	1	1	0	
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	
128	64	32	16	8	4	2	1	214

Pro zjednodušení převodu z dvojkové soustavy do osmičkové a šestnáctkové si můžeme uvědomit, že hodnotu čísla v osmičkové soustavě tvoří vždy trojice dvojkových čísel a hodnotu čísla v šestnáctkové soustavě vždy čtveřice dvojkových čísel. Po rozdělení binárního čísla na trojice (respektive čtveřice) stačí převést dvojková čísla na číslíky vyjadřující jednotlivé řády osmičkového (šestnáctkového) čísla:

$(111000)_2$	111	000	$(111000)_2$	0011	1000
$(70)_8$	7	0	$(38)_{16}$	3	8

## Kódování informací

**Kód** představuje pravidlo, jak převést jednu množinu znaků na jinou; např. jednotlivým alfanumerickým znakům se přiřadí čísla (viz kódování ASCII), *morseovka*, *Braillovo slepecké písmo* atd. Množina všech přípustných znaků se nazývá **abeceda** (např. Morseho telegrafická

abeceda). Při zpracování informací pomocí digitální techniky se používá **binární abeceda**, která obsahuje pouze dva znaky (0,1). S kódy v různých podobách se v informatice setkáváme na každém kroku. Zajímavými příklady grafických kódů jsou **čárové kódy** a **QR kódy**.

### ČÁROVÝ KÓD

Čárový kód se skládá z tmavých čar a ze světlých mezer, které se čtou pomocí specializovaných čteček - **snímačů čárových kódů** (laserových i digitálních). Některé typy čárových kódů mohou kódovat pouze číslice, jiné i písmena a speciální znaky. **Jednodimenzionální kódy** (1D) mají omezenou kapacitu a obvykle kódují numerický nebo alfanumerický řetězec, který je klíčem k identifikaci označeného předmětu do nějaké externí databáze. **Dvoudimenzionální kódy** (2D) vzhledem k vyšší kapacitě obvykle obsahují veškerou potřebnou informaci o označeném předmětu. Nejběžnější čárové kódy, používané pro zboží v obchodech, mají označení **EAN 13** a **EAN 8**.



### QR KÓD

Zkratka QR pochází z anglického „*quick response*“ (česky „rychlá odpověď“). V praxi stačí vyfotit QR kód mobilním telefonem a **QR čtečka** poté rozšifruje znaky kódu a přeloží je na žádané informace (text, vizitku, internetový odkaz). K vytvoření vlastního QR kódu je možné využít některý z online generátorů. QR kód je tvořen **geometrickou vrstvou** (slouží k přesné lokalizaci geometrických pozic, kde mají být čteny informační bity) a **informační vrstvou** (každý černý bod kóduje binární jedničku a bílý bod binární nulu). Dokáže pracovat také s určitým zkreslením a kompenzovat ztrátu části dat. QR kód je možné využít v řadě oblastí – od propagace, přes poskytování rychlých informací v galeriích či muzeích až po moderní platby.





**Soubor** (*file*) je množina souvisejících dat uložených na diskové jednotce. Název souboru je obvykle tvořen jménem (ve Windows až 254 znaků) a příponou, která naznačuje **typ souboru**. K dalším vlastnostem souboru patří jeho velikost, autor (vlastník), datum vytvoření, datum změny, atributy, přístupová práva apod.

K přehlednější práci se soubory slouží systém **adresářů** či složek (*directories, folders*). Adresáře tvoří na disku typicky hierarchickou strukturu (*tree - strom*); základem je tzv. **kořenový adresář** (*root*) označený symbolem „/“ (Unix, Linux), nebo „\“ (Windows). Každý adresář může obsahovat soubory i **podadresáře** (*subdirectories*).

**Cesta** (*path*) vyjadřuje umístění souboru vždy směrem od nejvyššího adresáře: C:\XAMPP\mysql\bin\mysql.exe. Pro operace se soubory využíváme někdy **zástupné znaky** (*wildcards*):

- symbol \* nahrazuje v názvu libovolný počet znaků (např. \*.\*);
- symbol ? nahrazuje jeden libovolný znak (např. s?s.txt).

**Programové soubory** obsahují instrukce, podle nichž procesor ve spolupráci s dalšími hardwarovými komponentami plní konkrétní úlohy. Mohou být uloženy v již zkompilované binární podobě (nejčastěji ve **strojovém kódu** pro danou platformu), nebo ve **zdrojovém kódu**, ob-

sahujícím příkazy určitého programovacího jazyka. Mezi programové soubory můžeme zařadit také **skripty**, soubory instrukcí v textové podobě, které ke svému spuštění vyžadují speciální program - **interpret**.

Větší programy jsou tvořeny **programovými knihovnami** (*runtime libraries*), speciálními soubory s připravenými funkcemi, které programy využívají pro svůj běh. Ve Windows to jsou např. soubory *DRV* a *SYS* (obsahující zejména ovladače čili drivery, ale i systémové funkce), nebo také **DLL** (*dynamic linking library* - dynamicky linkované knihovny), které mohou být podle potřeby za běhu (dynamicky) připojeny k jednomu i více programům.

Některé soubory s tzv. **spustitelnou příponou** (např. .exe a .com ve Windows, .sh v Linuxu) slouží ke spuštění programu - aktivaci procesu v rámci operačního systému. Za **rezidentní programy** (služby, démony) označujeme ty procesy, které jsou spuštěny automaticky po iniciaci operačního systému (např. antivirové programy).

**Datové soubory** obsahují data různého charakteru (číselné, textové, grafické, zvukové údaje). K jejich zpracování se používají konkrétní programy, a proto jsou často přípony těchto datových souborů asociovány (propojeny) s určitou aplikací (např. DOCX s programem MS Word, nebo XLSX s programem MS Excel).

Programové vybavení (*software*) je podmínkou využití technického vybavení (*hardware*). V raných dobách počítačové éry byly programy vytvářeny vždy pro konkrétní hardware. V okamžiku, kdy počítače začaly být produkovány ve větších sériích, objevila se i potřeba vývoje speciálních **systémových programů**, které by obstaraly základní operace s hardwarem a nabídly prostředí pro běh aplikací, tj. programů určených pro konkrétní účely.

Základním programovým vybavením počítačů (a jim podobných zařízení) jsou **operační systémy** (*OS - Operation System*). Jejich nejdůležitější funkcí je zajištění řízení hardware. To má na starosti **jádro systému**, které tvoří subsystémy pro správu procesoru, paměti, diskových médií a dalších zařízení. Jádro pro své operace vyžaduje instalaci ovladačů jednotlivých zařízení (*drivers*) a s jejich pomocí zpřístupňuje technické vybavení běžícím procesům.

Nezbytnou součástí operačních systémů je **aplikační rozhraní** (*API - Application Programming Interface*), které nabízí programátorům prostředky pro vývoj aplikací pro daný systém. **Uživatelské rozhraní** slouží k ovládání OS a k spouštění aplikací. Zatímco starší OS se ovládaly častěji prostřednictvím **příkazového řádku** (*CLI - Command Line Interface*), moderní systémy poskytují **grafické uživatelské rozhraní** (*GUI - Graphic User Interface*) založené na oknech a grafických prvcích.

Na počátku vývoje moderních OS byl víceúčelový a víceuživatelský systém **UNIX**, který se od 70. let používal v sálových počítačích i minipočítačích. Jeho principy využily i další OS. V 80. letech minulého století si vzájemně konkurovaly systémy firmy **Microsoft** (**MS DOS**), instalované na počítačích IBM PC, a firmy **Apple** (**Mac OS**), která rozvinula myšlenky vývojářů společnosti **Xerox** a začala jako první používat grafické uživatelské rozhraní. To se stalo také základem **MS Windows**, jenž hrál prim především na přelomu tisíciletí. V té době už získával stále větší popularitu volně šiřitelný systém **Linux**, pojmenovaný podle svého tvůrce **Linuse Torvaldse** a staršího vzoru UNIXu. Zatímco operační systémy Windows dominují především na osobních počítačích v domácnostech i kancelářích, různé linuxové distribuce jsou velmi často instalovány na síťových serverech.

Současné OS rozdělujeme nejčastěji podle typů zařízení, pro která jsou určeny. Existují **systémy pro síťové servery** (např. *MS Windows 2012 Server, Linux, Unix*), **pro běžné osobní počítače** - desktopy, notebooky, netbooky (např. *MS Windows, Linux Ubuntu, OS X*), **pro mobilní zařízení** (*Android, Symbian, Windows Phone, iOS*) i speciální **RTOS** (*Real-Time Operating System - systém reálného času*), které zajišťují provoz „chytrých“ zařízení, jakými jsou robotické kamery využívané pro sportovní přenosy, měřicí přístroje na orbitálních stanicích nebo jednotky pro řízení různých průmyslových procesů. Zatímco u desktopového systému je možné pozdržet dokončení některých operací a provádět paralelně jiné operace, v případě RTOS je nezbytné provádět operace bez prodlevy a s vysokou spolehlivostí v reálném čase.

K systémovému software můžeme kromě operačních systémů zařadit ovladače hardwarových zařízení, firmware a aspoň zčásti i tzv. utility.

**Ovladače zařízení** (*device drivers*). Speciální programy, které zajišťují komunikaci OS s určitým technickým vybavením. Bez potřebného ovladače nemůže zařízení správně fungovat. Standardní ovladače pro běžná zařízení, jakými jsou flash disk, myš, klávesnice nebo digitální fotoaparáty, bývají součástí moderních OS. Instalaci zařízení ve Windows usnadňuje standard **Plug and Play** (*PnP - „připoj a hrej“*), který zajistí automatickou detekci nově připojeného hardware a případně i automatické vyhledání potřebného ovladače. Není-li driver nalezen přímo v OS, musí být použit externí instalátor ovladače z disku dodaného se zařízením (např. z instalačního CD k tiskárně), nebo stažený z internetových stránek, nejlépe přímo od výrobce. Nevhodně zvolený ovladač může v krajním případě způsobit nefunkčnost zařízení, někdy i celého OS.

**Firmware**. Programové vybavení, které slouží pro řízení nějakého vestavěného systému (*embedded system*). Firmware řídí například kalkulačky, počítačo-

vé komponenty (pevný disk, router...), mobilní telefon, digitální fotoaparát atd. Dříve byla aktualizace firmwaru realizována nahrazením paměťového média obsahujícího firmware (např. výměna ROM paměti), dnes může být novější verze firmware přepsána díky používaným flash pamětím. Základním firmwarem v počítači je **BIOS** (*Basic Output-Input System*), který zajišťuje především start počítače a umožňuje konfiguraci základní desky i připojeného hardware.

**Utility**. Jsou menší pomocné programy, které umožňují vykonávat servisní operace potřebné k fungování počítače. Patří k nim komprimační programy, diagnostické a testovací programy, utility pro kontrolu a optimalizaci operačního systému, různé konverzní (převodní) programy, které umí přehledně zobrazit obrázky, převést je z jednoho formátu na jiný atd. Řada utilit je integrální součástí OS. Příkladem může být **Správce úloh** ve Windows (*Task Manager*), který umožňuje nejen správu spuštěných programů a služeb, ale i sledování využití systémových zdrojů (procesoru, paměti). Dalšími užitečnými utilitami jsou nástroje pro defragmentaci (*defrag*) nebo kontrolu disku (*chkdsk*). Některé utility lze spouštět z příkazového řádku (*ping, ipconfig* apod.).

Applikace jsou uživatelské programy určené pro jeden nebo více specifických účelů. Pro spouštění aplikací je nezbytná přítomnost základního software - operačního systému.

### ZÁKLADNÍ SKUPINY APLIKACÍ

**Kancelářské balíky.** Symbolický „balík“ programů obvykle tvoří textový procesor, tabulkový procesor, nástroj na tvorbu prezentací, databázový systém, někdy i groupwarové nástroje (systémy organizování času, zajišťování schůzek), grafické editory apod. Vedle komerčních *Microsoft Office* či *Apple iWork* existují také balíky *OpenOffice* nebo *WPS Office*, dostupné zdarma a pro řadu platform.

**Grafické aplikace.** Oblast grafiky je velmi široká; grafické programy umožňují vytvoření grafických dokumentů přímo v počítači, skenování papírových předloh i zachycení obrazu digitálním fotoaparátem. Dominantní postavení v této oblasti zaujímá profesionální software firmy *Adobe* (např. *Adobe Photoshop*, *Adobe Illustrator* aj.). Mezi oblíbené multiplatformní open-source grafické aplikace patří např. rastrový editor *GIMP* nebo vektorový *Inkscape*. Svěbytné oblasti počítačové grafiky tvoří programy pro tvorbu animací, rozsáhlé konstrukční systémy **CAD** (*Computer Aided Design*) nebo programy **DTP** (*Desktop Publishing*) určené pro tvorbu tiskovin.

**Multimediální programy.** Umožňují vytváření, zpracování a prezentaci multimediálních, tedy především audiovizuálních dat na počítači. Kromě nejrozšířenějších multimediálních přehrávačů zde můžeme zařadit řadu programů pro editaci digitálního zvuku nebo videa, programy pro tvorbu multimediálních prezentací, výukové programy, ale také rozsáhlou oblast počítačových her.

**Programy pro komunikaci v počítačové síti.** Slouží pro správu počítačových sítí, vzdálené řízení počítače, přenos a přístup ke vzdáleným souborům a vzdálený tisk, pro připojení vnitřní podnikové sítě (intranet) na síť celosvětovou (Internet) přes poskytovatele Internetu. K základním uživatelským (klientským) aplikacím patří webové prohlížeče, klienty elektronické pošty a groupware (software pro skupinovou práci), prostředky online komunikace (*IM* - *Instant Messaging*) a jiné.

**Informační a databázové systémy.** Informační systémy (*IS* - *information systems*) představují komplexní programové vybavení určité instituce nebo firmy. Základem IS jsou obvykle tzv. *datové sklady* (*data warehouse*) založené na databázových systémech. Podnikové informační systémy bývají označovány zkratkou *ERP* (*Enterprise Resource Planning*) a jsou tvořeny celou řadou dílčích programových modulů (např. moduly pro výrobu, logistiku, správu majetku, účetnictví ...). K lídrům v oblasti informačních i databázových systémů patří velké firmy jako *Oracle*, *Microsoft*, *IBM*, *SAP*. Kromě komplexních IS jsou vyvíjeny mnohé specializované aplikace pro podporu výrobních procesů, ekonomických transakcí a nejrůznějších jiných pracovních činností.

**Malware a antivirové programy.** Zatímco malware je počítačový program určený ke vniknutí nebo poškození počítačového systému, cílem antivirových programů (např. *AVG*, *AVAST*, *ESET*, *Kaspersky Antivirus* atd.) je chránit počítačové systémy před tímto nežádoucím softwarem.

**Integrovaná vývojová prostředí.** Obvykle jsou označovány zkratkou *IDE* (*Integrated Development Environment*) a používají je programátoři k vývoji aplikací. Tradičně je tvoří editor s různými pomůckami pro psaní programového kódu, ladicí prostředky, překladače a další nástroje zefektivňující práci vývojářů.

**Instalace programu.** Program, který si koupíme, dostaneme na CD/DVD disku, nebo si ho stáhneme z Internetu. Po vložení instalačního CD/DVD disku se většinou okamžitě spustí instalační program (díky souboru *autorun.inf*), v případě softwaru staženého z Internetu musíme instalační program spustit sami (ve Windows nejčastěji poklepnutím na soubory *install.exe* či *setup.exe*). Průvodce instalací (*wizzard*) si často vyžádá sériové číslo a souhlas s licenčními podmínkami, zeptá se, kam (do jaké složky na disku) se má program uložit a kam se má umístit zástupce programu, případně žádá další informace. Potom instalační program nahraje všechny soubory, ze kterých se program skládá, na pevný disk počítače, provede nutnou konfiguraci (v systémovém registru Windows, konfiguračních souborech) a nakonec vytvoří zástupce programu pro jeho snadné spuštění. K instalaci většiny programů je nutné mít práva správce.

**Odinstalace programu.** Úplné odebrání programu bývá paradoxně často složitější, zvláště v systémech Windows. Při instalaci si program nahraje své komponenty do mnoha složek včetně systémových, sdílí některé soubory s jinými programy (dynamické knihovny DLL), vytvoří záznamy v tzv. registru, při svém běhu vytváří pomocné soubory. Program, zajišťující odinstalování aplikace, často neví přesně, kde všude jsou programové soubory uloženy a všechny je nesmaže. V řadě případů tak dochází k hromadění zbytečných a nepoužívaných souborů. Pro jejich odstranění lze používat některé speciální utility – např. *RegCleaner* (program pro čištění registrů). Při podobných operacích je však nutné postupovat velmi obezřetně, aby nedošlo k odstranění některé důležité součásti systému nebo používaného programu! V linuxových systémech je lépe vyřešeno oddělení systémových a uživatelských souborů, úplné odstranění nepotřebného programu je proto pravděpodobnější.

**Lokalizace.** Lokalizace je překlad programu, v našem případě do češtiny. Většina významných softwarových firem pochází z anglicky mluvících zemí a originály programů tedy existují v angličtině. Dnes je však zcela běžné, že tyto firmy mají místní zastoupení, která zajišťují prodej programů, ale také je před prodejem lokalizují - přeloží do češtiny. Lokalizace je drahá a časově náročná činnost, protože se musí přeložit nejen volby nabídek a obsah dialogových oken, ale také nápověda k programu a návod k programu (manuál), případně vytvořit dělicí slovník pro české dělení slov a český korektor pravopisu.

**Platforma a kompatibilita.** V souvislosti s určitým typem OS i typem procesoru používáme pojem **platforma** - např. Windows, Linux, OS X. Používaný program musí být určen pro operační systém, který je na počítači nainstalován; pod jiným systémem většinou nepracuje (např. program určený pro MS

Windows nebude pracovat pod Linuxem a naopak). Mnoho programů, zvláště z oblasti svobodného software (*free software*), má proto verze určené pro několik operačních systémů – jsou **multiplatformní**. Problémy se spuštěním programu však mohou nastat i na jedné platformě v souvislosti s různými verzemi OS i rozdíly mezi 32bitovými a 64bitovými systémy (například nový program určený pro MS Windows může spolehlivě fungovat v novějších verzích Windows, ale nemusí pracovat pod starší verzí stejného OS). Při pořizování programu si proto vždy musíme ověřit, zda je **kompatibilní** (ve funkční shodě) s námi používaným operačním systémem. V nových systémech Windows je v případě problémů možné využít nastavení na kartě *Kompatibilita*, která je součástí panelu *Vlastností* pro zástupce dané aplikace.

**Verze programů.** U počítačových programů je důležitá jeho verze, která se označuje číslem za názvem programu. Nový program může být například označen jako verze 1.0. Některé dodatečně zjištěné dílčí nedostatky nebo bezpečnostní problémy řeší *aktualizace programu* (dnes nejčastěji přes Internet), **update**. Aktualizace jsou zakázáním, kteří si zakoupili určitou verzi programu, obvykle poskytovány bezplatně. Pro přehlednost jsou označovány buď datem vydání, nebo symbolickým číselným vyjádřením podverze – např. 1.1 nebo 1.23 apod. Povyšení na novou verzi (např. 2.0), **upgrade**, přináší významnější změny ve vývoji programu a v případě komerčního software je třeba vyšší verzi znovu zaplatit. Bývá pravidlem, že nové verze mají nové funkce a lepší ovládání, ale také vyšší nároky na výkon počítače. Před provedením upgrade je tedy nutné zjistit doporučenou (případně minimální) konfiguraci počítače, aby bylo možné program na daném počítači možné vůbec používat. Opačný postup, tedy použití nižší verze programu (např. kvůli nekompatibilitě s instalovanými OS nebo nedostatečnou konfigurací počítače), se nazývá **downgrade**.

**Registrace programu** je zaevidování jeho uživatele do databáze výrobce programu. Registrace je nepovinná a přináší většinou několik výhod:

- nárok na upgrade programu za sníženou cenu; uživatel se často včas dozví o časově omezených nabídkách výhodného nákupu upgradu nebo dalších programů od stejné firmy;
- někteří výrobci programů posílají svým registrovaným uživatelům materiály (časopisy, prospekty), kde se dozvíte zkušenosti jiných uživatelů s programem, bývají zde informace o nových verzích programu, případně o vzniklých doplňcích daného programu;
- u některých programů po registraci vzniká nárok na využití zákaznické linky v případě potíží s programem.